

Reconstrucción tomográfica con métodos estocásticos

R. Barbuzza¹, M. Vénere^{1,2}, A. Clausse^{1,2,3}, C. Moreno⁴

¹ *PLADEMA-ISISTAN, Universidad Nacional del Centro, 7000 Tandil, Argentina*
Tel; (02293) 442202, rbarbu@exa.unicen.edu.ar

² también *Comisión Nacional de Energía Atómica*

³ también *CONICET*

⁴ *PLADEMA y INFIP, Universidad de Buenos Aires, 1429 Buenos Aires, Argentina*

Resumen

En el presente trabajo se explora las posibilidades de generar tomografías con métodos no convencionales. En primer lugar se analiza el comportamiento del método Monte Carlo como algoritmo de reconstrucción tomográfica proponiéndose distintas estrategias y estudiándose para cada caso la calidad de las distintas reconstrucciones. Los casos tratados en primer lugar son reconstrucciones de objetos simples a partir de proyecciones obtenidas por simulación computacional de forma de poder evaluar la calidad. Finalmente, se muestra una aplicación en un caso real, donde las proyecciones son obtenidas mediante el disparo de un dispositivo Plasma Focus sobre un objeto metálico.

Palabras claves: tomografía computada, reconstrucción tomográfica, Monte Carlo

1 Introducción

Una tomografía computada es una imagen tridimensional digital construida a partir de un cierto número de proyecciones radiográficas del objeto a diferentes ángulos. En la Figura 1 se muestra un corte bidimensional de un objeto elíptico, que es atravesado por un haz de radiación de intensidad I_0 , generando detrás un perfil proyectado I . El perfil está relacionado con la distribución local de atenuación dentro del objeto $f(x,y)$ de acuerdo a:

$$g(s, \mathbf{q}) = \ln \left(\frac{I}{I_0} \right) = \mathbf{R} f(x, y) \quad (1)$$

donde \mathbf{R} es la transformada de Radon, \mathbf{q} es el ángulo de proyección, s corresponde a cada rayo que atraviesa al objeto en dirección perpendicular a \mathbf{q} .

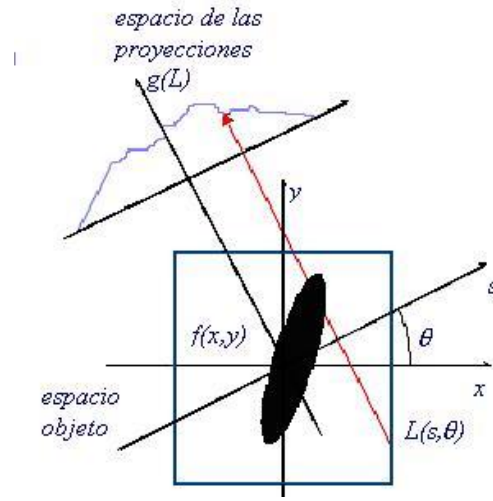


Figura 1. Proyección de una imagen

La tomografía computada consiste en resolver el problema inverso de hallar $f(x,y)$ a partir de $g(s, \mathbf{q})$ [6]. Existen diferentes métodos basados en modelos determinísticos o probabilísticos que permiten encontrar la imagen original resolviendo la Ec. (1). Los primeros han sido bastante usados en distintas aplicaciones comerciales, y trabajan bien cuando se cuenta con un conjunto completo de proyecciones y que además no tienen alto nivel de ruido. Entre los más conocidos se pueden citar ART, Filtered Backprojection, Convolution Backprojection, de Fourier, entre otros[6]. Actualmente, se ha comenzado a utilizar algoritmos alternativos basados en modelos estocásticos ya que estos han logrado mejorar la calidad de la reconstrucción en aplicaciones que cuentan con un conjunto limitado de proyecciones como también cuando las mismas tienen ciertas imperfecciones (Ej. ruido). La ventaja de estos algoritmos es que permiten buscar aleatoriamente una solución que mejor aparezca el comportamiento de datos probabilísticos. Además, estos modelos pueden mejorar substancialmente la calidad de la reconstrucción incorporando información a priori acerca de sistema de medición (Ej. Ponderación de la calidad de las distintas proyecciones) y del objeto que se quiere reconstruir (Ej. ciertas formas específicas como fisuras en componentes mecánicos o tumores en medicina). Los métodos probabilísticos más frecuentemente usados son Bayesian y ML-EM [2][3][5]. En este trabajo, se propone aplicar el método probabilístico Monte Carlo, el cual es comúnmente usado para aproximar la solución en problemas de optimización en general.

2 Método de Monte Carlo para la reconstrucción tomográfica

Monte Carlo realiza una búsqueda estocástica en el dominio que contiene la solución de un problema matemático dado, guiado por un criterio de selección que asegura la tendencia hacia la solución verdadera. En el caso de la reconstrucción tomográfica, Monte Carlo propone aleatoriamente funciones $f(x,y)$, hasta encontrar la que satisface la Ec. (1). El muestreo se realiza siguiendo una secuencia de funciones $f_i(x,y)$ que difieren unas de otras en cambios pequeños: se varía el tono de algunos píxeles. Llamamos $f_0(x,y)$ a la imagen inicial de esta secuencia. Para asegurar que la búsqueda tienda a la solución, se establece un criterio de aceptación de los cambios propuestos, que indica el grado de satisfacción de la Ec. (1). Para ello, se busca minimizar una función de error, que en este caso expresa la diferencia entre las proyecciones originales $g(s,\mathbf{q})$ y las proyecciones $g_i(s,\mathbf{q})$ de la imagen propuesta $f_i(x,y)$ luego de cada mutación. La función de error propuesta es la suma de los errores individuales de cada proyección según:

$$\mathbf{e}_i = \sum_{\mathbf{q}} (g(\mathbf{q}) - g_i(\mathbf{q}))^2 \quad (2)$$

donde $g_i(\mathbf{q})$ es la proyección en la dirección \mathbf{q} de la $f_i(x,y)$ y $g(\mathbf{q})$ es la proyección real en la dirección \mathbf{q} . En la ecuación la suma se realiza para todos los rayos de cada proyección.

Es claro que el mínimo de esta función se da cuando $g(s,\mathbf{q})=g_i(s,\mathbf{q})$ para cada ángulo \mathbf{q} dato, convergiendo así en la imagen original buscada. Luego, se considera que la i -ésima mutación es adecuada si el indicador de error \mathbf{e}_i correspondiente a la nueva imagen $f_i(x,y)$ es menor que el de la última mutación aceptada. Las alteraciones a la imagen se realizan hasta que se llega a un valor de error aceptable. La Figura 2 muestra el esquema básico de este algoritmo.

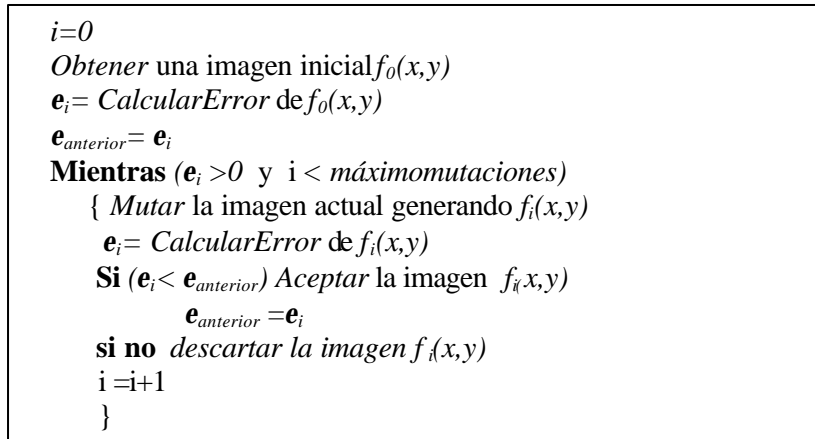


Figura 2. Esquema básico del algoritmo Monte Carlo

3 Algoritmos analizados

El procedimiento de mutación genera una nueva imagen $f_i(x,y)$ alterando un conjunto de píxeles víctimas de la imagen previa. Un aspecto importante de cada mutación es determinar el número de píxeles que serán alterados simultáneamente y la intensidad que se asignará a cada uno de ellos. A continuación se describen dos métodos de selección de píxeles víctimas, y dos métodos de asignación de tonalidad a los píxeles seleccionados en cada mutación.

3.1 Métodos de selección de píxeles víctimas

Los píxeles candidatos a ser seleccionados serán aquellos que pertenecen con un alto grado de certeza al objeto que se quiere reconstruir. Es decir, los píxeles víctimas resultan de la intersección de rayos tal que $g(s, \mathbf{q}_n) > 0$ para cualquier \mathbf{q}_n (ubicación del objeto). En el ejemplo de la Figura 3 se muestran tres proyecciones en $\mathbf{q}_1, \mathbf{q}_2$ y \mathbf{q}_3 de una elipse. El punto 1 de esta figura es un punto válido ya que pertenece a la zona de intersección donde $g(s, \mathbf{q}_1) > 0$, $g(s, \mathbf{q}_2) > 0$ y $g(s, \mathbf{q}_3) > 0$, mientras que el punto 2 en el rayo que lo atraviesa en la dirección de la proyección \mathbf{q}_3 tiene $g(s, \mathbf{q}_3) = 0$ y por lo tanto no se considera válido para el objeto que se quiere reconstruir.

Se consideraron dos métodos de selección de píxeles víctimas que llamaremos *Pithecanthropus* y *Cro-Magnon*[4]. Con *Pithecanthropus* todos los píxeles válidos tienen la misma probabilidad de ser seleccionados. Para ello, se toman dos proyecciones en forma aleatoria de la imagen original $g(s_1, \mathbf{q}_1)$, $g(s_2, \mathbf{q}_2)$. Luego se sortea con distribución uniforme dos rayos s_1 y s_2 entre los que satisfacen $g(s_1, \mathbf{q}_1) > 0$ y $g(s_2, \mathbf{q}_2) > 0$. Se calcula el punto víctima como la intersección entre s_1 y s_2 , el cual se acepta si se encuentra en la zona de atenuación de todas las proyecciones, en caso contrario se descarta y se repite el procedimiento.

El procedimiento *Cro-Magnon* es igual que *Pithecanthropus*, pero los rayos s_1 y s_2 se sortean con probabilidad proporcional a $|g(s, \mathbf{q}_1) - g_i(s, \mathbf{q}_1)|$ y $|g(s, \mathbf{q}_2) - g_i(s, \mathbf{q}_2)|$, donde $g_i(s, \mathbf{q})$ es la proyección de la imagen reconstruida hasta ese momento y $g(s, \mathbf{q})$ es la proyección original en la misma dirección. De esta manera se favorecen los puntos que requieren mayor corrección.

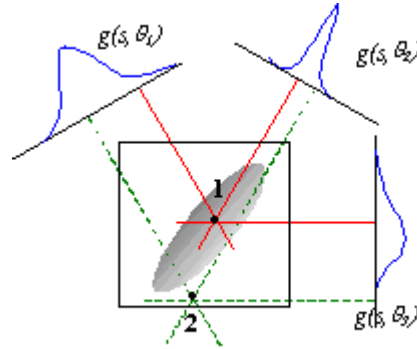


Figura 3. 1: Punto elegible , 2: Punto no elegible

3.2 Tonalidad de los píxeles víctimas

Un aspecto importante de estas técnicas es el rango de variación de intensidad con que se ajustará la intensidad los píxeles víctimas. En el caso de aplicar cambios pequeños, no alterarán demasiado a la imagen, y la convergencia será demasiado lenta. Por el contrario, si se aplican cambios bruscos de tonalidad, la imagen generada tendrá un aspecto poco uniforme afectando la calidad de la misma. Se implementaron dos estrategias para ajustar la intensidad: por *Asignación* o por *Negociación*.

Con *Asignación*, se selecciona un único píxel al cual se le aplica un ajuste aleatorio de tonalidad según:

$$\text{TonoPixelV\acute{ic}tima} = \text{TonoPixelV\acute{ic}tima} + \text{AjusteGris} \quad (3)$$

donde *AjusteGris* es un valor generado aleatoriamente dentro del rango establecido por el usuario.

El método *Negociación*, transfiere tonalidad entre dos píxeles dispuestos a negociar. La ventaja de esta técnica con respecto al método de *Asignación* es que mantiene constante el valor de intensidad total de la imagen que se quiere reconstruir. Este valor aporta información adicional al proceso de reconstrucción y se puede determinar a partir de las proyecciones originales $g(s, \mathbf{q})$. En la negociación a uno de los píxeles involucrados se le suma y a otro se le resta el valor de intercambio manteniendo invariante la intensidad total, según se muestra en la ecuación:

$$\begin{aligned} \text{TonoPixelVictima1} &= \text{TonoPixelV\acute{ic}tima1} + \text{ValorIntercambio} \\ \text{TonoPixelV\acute{ic}tima2} &= \text{TonoPixelV\acute{ic}tima2} - \text{ValorIntercambio} \end{aligned} \quad (4)$$

donde *ValorIntercambio* es un valor establecido a lo largo de la reconstrucción. Al inicio los píxeles pueden negociar por valores grandes, ya que la imagen en reconstrucción necesita acercarse más la solución. Por el contrario, cuando la imagen reconstruida necesita pocos cambios, las negociaciones sólo se concretarán por valores más pequeños.

3.3 Metrópolis

Como se describió en la Figura 2, en la reconstrucción con Monte Carlo a medida que se incrementan el número de mutaciones el error disminuye y la imagen reconstruida tiende a la imagen original. En ese caso siempre se seleccionan mutaciones cuyo error es menor que el de la última mutación aceptada ($\Delta\epsilon = \epsilon_i - \epsilon_{\text{anterior}} < 0$). Esta decisión puede ocasionar que el algoritmo quede atrapado durante la búsqueda en un mínimo local, que son configuraciones con $\epsilon > 0$ para las cuales cualquier cambio sólo aumenta ϵ . Por esta razón, se propuso una forma de evitar esta situación permitiendo incrementos ocasionales de la función de error, conocido como Metrópolis. Esta técnica se origina de la teoría de la mecánica estadística para la resolución de problemas de optimización [1]. Cuando se utiliza Metrópolis, si el cambio $\Delta\epsilon$ es negativo, la transición es aceptada incondicionalmente; caso contrario, se acepta con una probabilidad basada en la distribución de Boltzmann [1]:

$$p(\Delta\epsilon) \sim e^{-\frac{\Delta\epsilon}{T}} \quad (5)$$

donde T se lo conoce como la temperatura y es un parámetro de control. La temperatura, puede ser un valor constante o puede decrementarse gradualmente a lo largo del algoritmo desde un valor suficientemente alto (todas las transiciones propuestas, tanto negativas como positivas son aceptadas) hacia un valor bajo (casi no ocurren cambios).

4 Estudio sobre un caso simple

Para estudiar el problema de la reconstrucción tomográfica, se analiza un ejemplo muy simple: una matriz de 4x4 píxeles de dos valores (blanco y negro) a partir de sus proyecciones a 0° y 90°. El análisis de cada una de las estrategias de reconstrucción sobre este ejemplo permitirá observar el comportamiento de las reconstrucciones en cada caso, y como se verá más adelante, anticipar los problemas de cada método.

La Figura 4 muestra la imagen que se supone desconocida y se deberá identificar con una función $f(x, y)$ a partir de las dos proyecciones. En este caso se restringe el experimento numérico a proyecciones sin ruido, por lo que se cumple que el valor de la proyección a 90° por rayo es igual a la suma de cada fila de la matriz, y la proyección a 0° por rayo es la suma de cada columna.

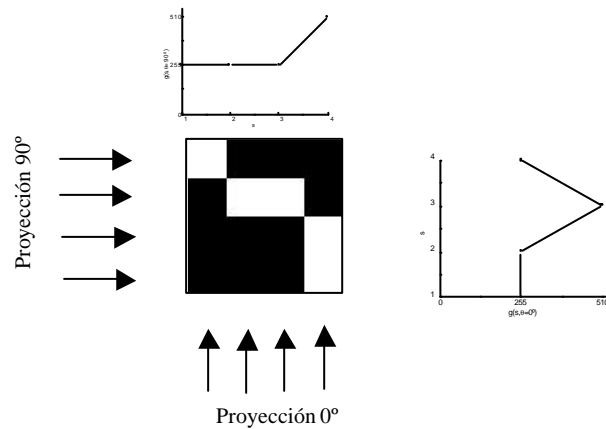


Figura 4. Un caso de estudio con una imagen blanco negro de 4x4 píxeles

El dominio de búsqueda para el problema de reconstrucción planteado es de 2^{16} (65536) posibles funciones $f(x,y)$. Se realizó un análisis evaluando la función de error ε definida en la Ec (2) a cada instancia del dominio. El resultado de este análisis muestra que para este problema, la solución no es única ya que existe un conjunto de 27 matrices $f(x,y)$ solución ($\varepsilon=0$). A la izquierda de la Figura 5 se muestra la distribución de error ε para las distintas funciones $f(x,y)$ del dominio. Como se puede observar, la mayoría de las instancias tiene función de error ε distinto de 0, con frecuencia máxima para instancias con error $\varepsilon=1249$. A la derecha de la Figura 5 se muestran las 27 matrices $f(x,y)$ del conjunto de soluciones ambiguas de este ejemplo.

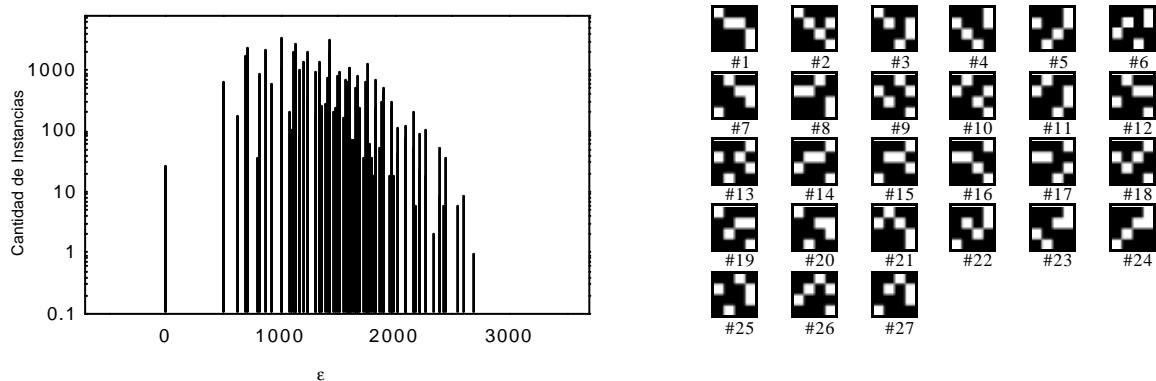


Figura 5. Distribución de error ε del dominio (izquierda) y conjunto de 27 instancias ambiguas con error $\varepsilon = 0$ (derecha)

Luego de estudiar el dominio de búsqueda de este problema, se usó Monte Carlo con distintos métodos para la búsqueda de una solución. A modo de ejemplo, en la Figura 6 se muestran dos reconstrucciones usando *Asignación*, es decir se selecciona un único píxel para cambiar la tonalidad (blanco a negro o viceversa), combinado con otras estrategias: *Pithecanthropus*+*Sin Metrópolis* o *Cro-Magnon*+*Sin Metrópolis*. En la gráfica se muestra la evolución del error ε a través de las mutaciones, y las imágenes aceptadas en cada mutación. Se puede ver como el método *Cro-Magnon* (negro) necesita menor cantidad de mutaciones para disminuir el error que *Pithecanthropus* (gris). Esto se debe a que *Cro-Magnon* selecciona píxeles que necesitan mayor corrección. Sin embargo, ambos métodos tienen el problema que caen en mínimos locales, ya que cualquier modificación que se realice a la última imagen no logra mejorar el último error obtenido.

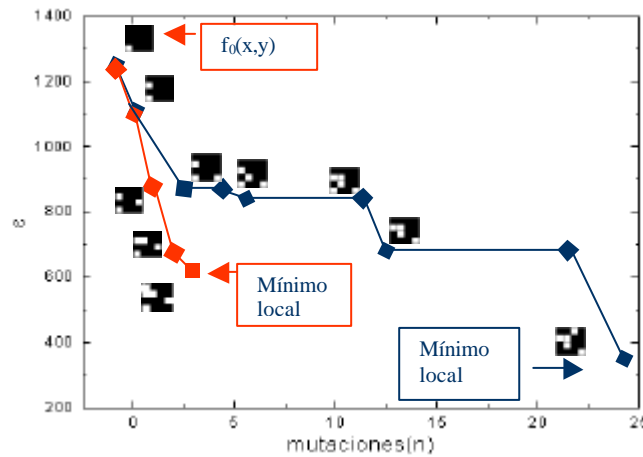


Figura 6. Uso de *Asignación* con dos posibles combinaciones *Pithecanthropus*+*SinMetrópolis* y *Cro-Magnon*+*SinMetrópolis*

Como se muestra en los ejemplos de reconstrucción de la Figura 6, el algoritmo puede caer en un mínimo local dependiendo de la trayectoria aleatoria de búsqueda. En ambas reconstrucciones, la trayectoria comienza con la misma imagen $f_0(x,y)$ que ha provocado que el algoritmo caiga en mínimos locales. Es claro que si se hubiese comenzado con otra imagen $f_0(x,y)$ puede darse que el algoritmo encuentre una solución.

Por esta razón, se estudió el comportamiento de las distintas estrategias de reconstrucción con Monte Carlo, comenzando con cualquiera de las posibles instancias $f_0(x,y)$ del dominio (65536 reconstrucciones). Se evaluó la performance de cada una de las estrategias, considerando el porcentaje de convergencia de las 65536 reconstrucciones, y el promedio de mutaciones necesarias para encontrar una solución (Tabla 1). Las tres primeras filas de esta tabla, corresponden a *Asignación* y las filas restantes a *Negociación*. Además se combinan estas estrategias con los distintos métodos de selección de píxeles *Pithecanthropus*, *Cro-Magnon*, con o sin *Metrópolis*. Dependiendo de cada reconstrucción, si Monte Carlo encuentra una solución, ésta puede coincidir con cualquiera de las 27 soluciones ambiguas de la Figura 5.

Para solucionar el problema de mínimos locales se aplicó la técnica de *Metrópolis*, Ec. (5), con $T=75$, la cual logró en el caso *Pithecanthropus*+*Metrópolis* el 100% de convergencia, sin embargo *Cro-Magnon*+*Metrópolis* no pudo alcanzar siempre una solución. Esto se debe a que *Cro-Magnon* puede quedar seleccionando un conjunto limitado de píxeles que requieran corrección y de esta manera no logra salir del mínimo local.

Como mejora de los métodos anteriores se utilizó la combinación *Cro-Magnon* al inicio de la reconstrucción y luego *Pithecanthropus* (*Cro-Magnon*+*Pithecanthropus*), la cual tuvo un comportamiento más adecuado que el uso individual de ellas.

Si se analizan los resultados obtenidos en la Tabla 1, se puede observar que la configuración *Asignación*+*Pithecanthropus*+*Sin Metrópolis* (primer fila) conduce a que sólo el 3% de las pruebas realizadas alcancen $e=0$. En el caso *Asignación*+*Cro-Magnon*+*Sin Metrópolis* (fila 2) se alcanza la convergencia más rápidamente que con *Pithecanthropus* ya que las mutaciones promedio disminuyeron de 20 a 7, pero al igual que en el caso anterior puede quedar atrapado en mínimos locales ya que sólo el 21% alcanzó la convergencia.

	Sin Metrópolis		Con Metrópolis	
	% conv.	Prom. mutac	% conv.	Prom. Mutac.
Asignación				
Pithecanthropus	3 %	20	100 %	161
Cro-Magnon	21 %	7	67 %	16
Cro-Magnon+Pithecanthropus	21 %	7	98 %	95
Negociación				
Pithecanthropus	99 %	40	99 %	45
Cro-Magnon	89 %	16	82 %	11
Cro-Magnon+Pithecanthropus	99 %	25	99 %	26

Tabla 1. Porcentaje de convergencia y promedio de iteraciones con distintas estrategias de reconstrucción

Cuando se agregó Metrópolis la combinación *Asignación+Pithecanthropus+Metrópolis* se alcanzó 100% de convergencia pero se incrementó notablemente el total de mutaciones promedio 20 a 161. La combinación *Pithecanthropus+Metrópolis* muestra que puede salir de mínimos locales y llegar siempre a una solución si se aumenta la cantidad de mutaciones. A diferencia *Cro-Magnon+Metrópolis* puede quedar atrapado en mínimos locales (sólo 67% convergencia). La mejor alternativa con *Asignación* fue *Cro-Magnon+Pithecanthropus+Metrópolis* que logró un 98% de convergencia con un promedio de 95 mutaciones, acelerando la convergencia con respecto usar sólo *Pithecanthropus*. En el experimento, se limitó la cantidad de mutaciones por reconstrucción en 500, por esta razón, algunas reconstrucciones no llegan a converger antes del límite.

Finalmente, se analizan los resultados obtenidos con el método de *Negociación* en la reconstrucción del ejemplo simple de la Figura 4. A diferencia del caso anterior, en cada mutación se seleccionaron dos píxeles víctimas para realizar la transferencia de tonalidad con valores 0 (negro) o 255 (blanco). Al igual que antes, se realizaron 65536 reconstrucciones comenzando con una imagen $f_0(x,y)$ diferente del dominio cada vez. Cada imagen inicial $f_0(x,y)$ tiene un determinado valor de intensidad total que puede coincidir o no con la intensidad buscada (en este caso se conoce por las proyecciones que deben existir 5 píxeles blancos como en la imagen original para comenzar la negociación).

Se puede observar en la Tabla 1 que la aplicación de la técnica de *Negociación* acelera notablemente la convergencia. Sin embargo, el uso de *Metrópolis* agregado a *Negociación* no logró mejorar los resultados. Por esta razón sólo se analizan los resultados de *Negociación sin Metrópolis*.

Negociación+Pithecanthropus alcanzó el 99% de convergencia con un promedio de 40 mutaciones. Nuevamente, *Negociación+Cro-Magnon* tiene el inconveniente que puede quedar seleccionando el mismo par de píxeles para negociar. Como antes, la ventaja de *Cro-Magnon* es que necesita menor cantidad de mutaciones por reconstrucción. En *Negociación*, la combinación *Negociación+Cro-Magnon+Pithecanthropus* resultó la mejor alternativa de reconstrucción con un 99% de convergencia y un promedio de 25 mutaciones.

Como resultado se puede decir que las combinaciones que resultaron con un buen porcentaje de convergencia y menor cantidad de mutaciones promedio son: *Asignación+Cro-Magnon+Pithecanthropus+Metrópolis* y *Negociación+Cro-Magnon+Pithecanthropus*. Pese a que esta última tiene una convergencia más rápida que la primera, se debe notar que el costo computacional de *Negociación* por mutación es mayor ya que necesita encontrar dos píxeles dispuestos a negociar, mientras *Asignación* cambia un único píxel por mutación.

5 Resultados con imágenes conocidas más complejas

Si bien con objetos pequeños como se mostró en la sección anterior se logró alcanzar el error mínimo, al considerar objetos más grandes la convergencia se tornó extremadamente lenta. Por esta razón, se utilizó un límite máximo de iteraciones aunque el error de la Ec. (2) no llega a alcanzar el mínimo absoluto. En la Figura 7 se muestra un ejemplo del comportamiento de Monte Carlo con imágenes de testeo con elipses heterogéneas de distinta excentricidad 0.3, 0.5 y 0.7 de tamaño 256x256 píxeles. En la primera columna se muestra la imagen original, en la segunda la imagen reconstruida con la técnica *Asignación+Cro-Magnon+Pithecanthropus+Metrópolis* y en la última columna la imagen reconstruida con *Negociación+Cro-Magnon+Pithecanthropus*. Los datos de las proyecciones corresponden a 8 ángulos: 0°, 30°, 60°, 75°, 90°, 105°, 120° y 150°, cada uno con 512 rayos paralelos. Como se puede ver el enfoque propuesto obtuvo buenas aproximaciones de las imágenes originales usando 50000 iteraciones. El desvío de tono utilizado en la Ec. (3) es [-80..80]. A la derecha de la figura, se muestra la tasa de error promedio ϵ/ϵ_0 (error actual respecto del error inicial) en función de las mutaciones.

Los resultados confirman los resultados obtenidos con la imagen sencilla del ejemplo de la Figura 4. Se puede observar que al inicio de la reconstrucción la tasa de error promedio con la estrategia *Negociación* desciende más abruptamente que con la variante *Asignación*. Sin embargo, hacia el final de la reconstrucción esta diferencia no es tan significativa.

Luego de 10000 iteraciones el error con *Negociación* alcanza a mejorar en promedio un 70% con respecto del error inicial y en el caso *Asignación* llega a mejorar un 50% respecto del error inicial. Esta diferencia entre los métodos se hace cada vez menor en las iteraciones siguientes, y se mantiene casi constante en 5% a partir de 35000 iteraciones. Hacia el final de la reconstrucción, en 50000 iteraciones, el error alcanzado por los métodos es muy similar alcanzando el 90% de mejora respecto del error inicial.

Con respecto a los tiempos de reconstrucción la variante con *Negociación* es más costosa ya que es necesario seleccionar dos píxeles dispuestos a negociar, mientras que con *Asignación* sólo se selecciona un píxel. En este ejemplo el algoritmo utilizó 0.012 seg. por iteración con *Negociación* y 0.009 seg. por iteración con *Asignación*, utilizando con un procesador Pentium 3 1.1 Ghz.

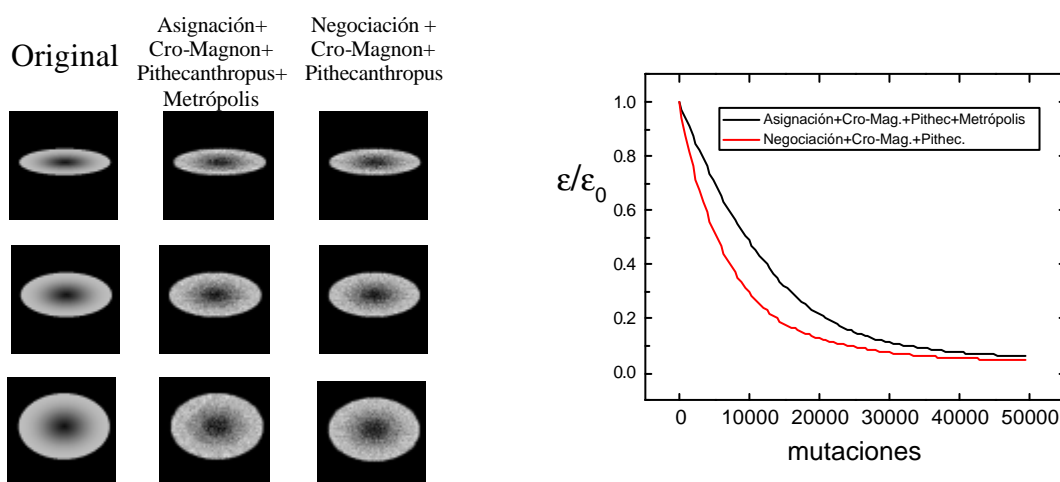


Figura 7. Elipses no homogéneas reconstruidas con Monte Carlo

6 Posprocesamiento para el suavizado de la imagen

En el caso específico de las imágenes obtenidas con la reconstrucción Monte Carlo, si bien han resultado de buena calidad presentan cierto aspecto granuloso. Las imágenes tomográficas en general se realizan sobre objetos los cuales se componen de un conjunto discreto de materiales cada uno con propiedades uniformes, o bien en el caso de medicina corresponden a distinta composición del cuerpo como hueso, tejido blando, etc. Como consecuencia, la reconstrucción ideal tendría que contener valores de píxeles vecinos que tengan más homogeneidad. Por este motivo, se implementaron tres filtros pasa bajo: Fourier bidimensional, Promedio y *FSLD* [10], con el objetivo de suavizar o para reducir el ruido de las imágenes obtenidas y substancialmente mejorar la calidad de la reconstrucción. Si bien se pudo apreciar que la imagen filtrada tiene mayor similitud con la imagen real (en este caso es conocida), se observó en oposición, que el error según la Ec. (2) aumentó. Para cuantificar la aproximación de la imagen reconstruida filtrada con la original se decidió comparar con el indicador I_i de error cuadrático medio entre dichas imágenes:

$$I_i = \sum_{x,y} (f(x,y) - f_i(x,y))^2 \quad (6)$$

En el ejemplo de la Figura 8, se muestra la imagen original, la imagen reconstruida con 50000 mutaciones sin filtrar, y tres imágenes suavizadas con los filtros mencionados. Para el ejemplo si no se aplica filtro el error entre proyecciones ϵ es 2486 y el error entre imágenes I es 5831. Se puede observar que mientras el error ϵ aumentó para cualquiera de las imágenes filtradas respecto a 2486, los errores I disminuyeron respecto a 5831. Este resultado muestra que para el caso de las imágenes tomográficas resulta beneficioso aplicar un filtro de suavizado hacia el final de la reconstrucción. El filtro de mejor comportamiento fue *FSLD*.

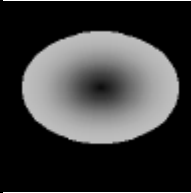
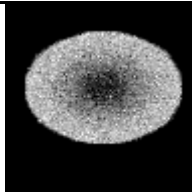
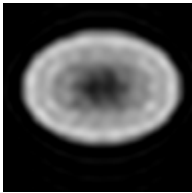
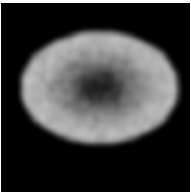
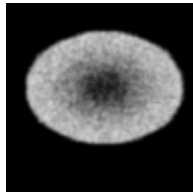
Original	Sin filtro	Con filtro		
		<i>Fourier</i>	<i>Promedio</i>	<i>FSLD</i>
				
ϵ proyecciones	2486	7077	7257	3073
I imágenes	5831	3300	3235	2699

Figura 8. Suavizados de la imagen reconstruida, error entre proyecciones y error entre imágenes de cada caso

7 Reconstrucción de un objeto real

En esta sección se presenta la aplicación del método Monte Carlo a un caso real de reconstrucción tomográfica 3D a partir de placas tomadas con un dispositivo Plasma Focus, diseñado y construido en PLADEMA-ISISTAN [7][8][9][11]. Para el experimento se usó un conector BNC de acero inoxidable, utilizado en el cableado de redes con cable coaxil, al cual se le tomaron 8 proyecciones, registrando la atenuación de intensidad de los rayos X en placas radiográficas. Antes de realizar la reconstrucción, se debieron tener en cuenta distintos tipos de errores que pueden introducirse en la captura de las proyecciones (ej. proyecciones rotadas o mal alineadas, precisión del ángulo, diferencia de intensidad, etc.). La Figura 9 muestra el objeto y las 8 proyecciones del conector para los ángulos fijados. La resolución en cada placa (tamaño píxel) es de 0.08 mm. en cada lado.

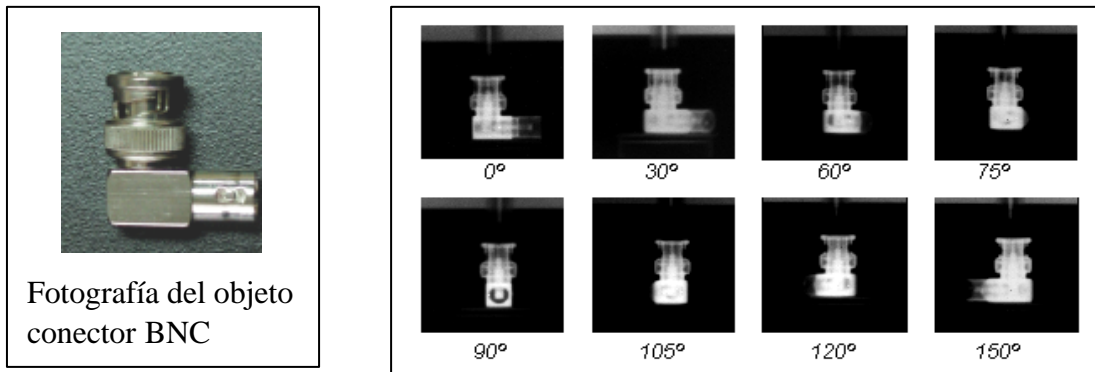


Figura 9. Objeto original (izquierda) y proyecciones obtenidas a 8 ángulos diferentes (derecha)

Para realizar la reconstrucción 3D del conector BNC, se aplicó la técnica de Monte Carlo por corte axial. La tomografía final resulta de la composición de todos los cortes axiales obtenidos (Figura 10). En este caso, se utilizó la técnica de *Asignación+Cro-Magnon+Pithecanthropus+Metrópolis*. Cabe aclarar que si bien las proyecciones reales tienen un ruido considerable, el algoritmo se adaptó fácilmente logrando una buena reconstrucción del objeto. La tomografía obtenida puede visualizarse, a través de los distintos cortes sagital, coronal o axial, como así también vistas en perspectivas (Figura 11).

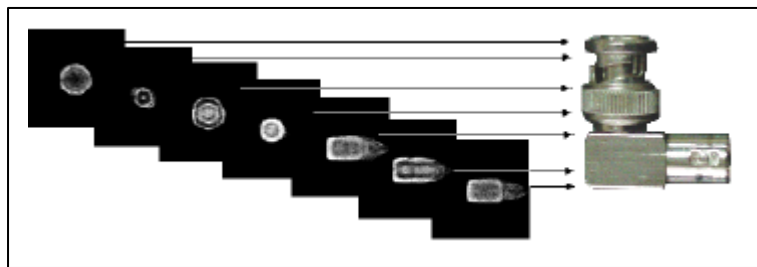


Figura 10. Distintos cortes axiales del conector reconstruidos con Monte Carlo

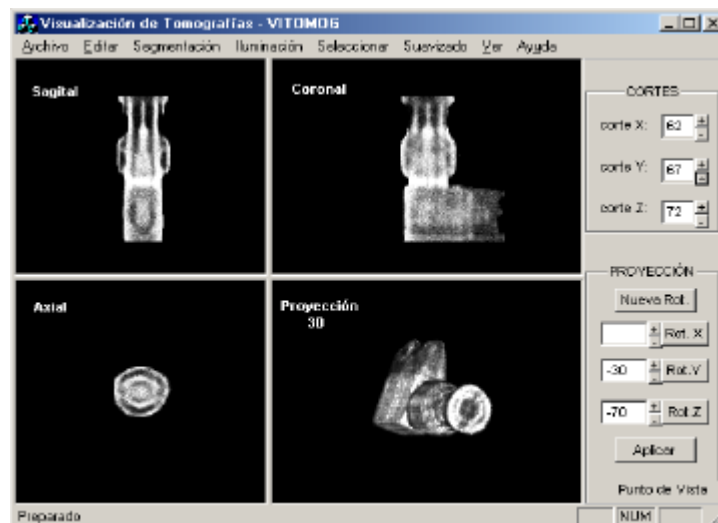


Figura 11. Visualización de tomografía y cortes del conector según los planos x,y,z

Con este método, el tiempo de reconstrucción promedio de los cortes axiales del conector para una resolución de 256x256 fue de 330 seg. El procesador utilizado en este caso fue un Pentium 3 1.1 Ghz. Este tiempo resulta comparable al obtenido con técnicas convencionales tal como ART.

8 Conclusiones

Con este estudio se mostró la factibilidad de realizar reconstrucciones tomográficas no convencionales de razonable calidad. El estudio de la reconstrucción de imágenes sencillas permitió observar el comportamiento de los algoritmos propuestos. La ventaja del método Monte Carlo es que permite incorporar fácilmente distintas estrategias de reconstrucción sin cambiar la estructura general del algoritmo, sin embargo al igual que otros métodos iterativos, Monte Carlo presenta el problema del tiempo computacional que insume para lograr la convergencia.

La calidad de las imágenes obtenidas fueron mejoradas con la aplicación de distintos filtros de suavizado, lo cual permitió acercarse a la solución del problema. En este sentido, sería interesante realizar en un futuro estudios de performance de algoritmos basados en distintos indicadores de calidad de imágenes obtenidas.

Actualmente, el empleo de tomografías se restringe casi exclusivamente al caso de medicina, siendo muy escasa su explotación en otras áreas. En este trabajo, se muestra que la técnica de reconstrucción implementada es especialmente útil para la visualización introspectiva de piezas metálicas generadas a partir de radiografías obtenidas con un dispositivo simple de emisión de radiación X con una fuente Plasma Focus. A partir de estos resultados, se prevé perfeccionar el procedimiento de medición de manera de que pueda obtenerse una tomografía de un objeto en movimiento.

9 Referencias

- [1] Chandler D., Introduction to modern statistical mechanics, Oxford University Press, (1987).
- [2] Frese T., C. Bouman, K. Sauer, "Discrete Multiscale Bayesian Image Reconstruction", Proc. thirty-second Conf. Signals, Sys. and Comp., Vol 2, pp. 1687-1691, Nov 1-4, Pacific Grove (1998).
- [3] Galigekere R., K. Wiesent, D. Holdsworth, "Techniques to alleviate the effects of view aliasing artefacts in computed tomography", Medical Physics, Vol 26, No 6, (1999).
- [4] Ghezzi R., B. Islas, M. del Fresno, R. Barbuzza y A. Clausse, "Algoritmos de Monte Carlo para la reconstrucción tomográfica", JAIIO 2000, 4 al 9 de sep, Tandil, Argentina, pp 25-32 (2000).
- [5] Green P. J., "Bayesian reconstruction from emission tomography data using a modified EM algorithm," IEEE Trans. on Medical Imaging, 9, pp 84-93 (1990).
- [6] Herman G., *Image Reconstruction from Projections. The Fundamental of Computerized Tomography*, Academic Press, London (1980).
- [7] Moreno C., A. Clausse, H. Bruzzone, "Ultrafast x-ray introspective imaging of metallic objects using a plasma focus device", Nukleonika V.46, pp. 33-34 (2001).
- [8] Moreno C., H. Bruzzone, J. Martínez, A. Clausse, "Conceptual engineering of plasma-focus thermonuclear pulsors", IEEE Transactions on Plasma Science, V.28, p. 1735-1741, (2000).
- [9] Moreno C., M. Vénere, R. Barbuzza, M. del Fresno, H. Bruzzone, P. Florido, J. Gonzalez, A. Clausse, "Industrial application of plasma focus radiation", Brazilian Journal of Physics, Vol 32, Nro 1, pp: 20-25, (2002).
- [10] Taubin G., "A signal processing approach to fair surface design", IBM Research Report, RX-19923(#88210) (1995).
- [11] Vénere M., C. Moreno, A. Clausse, R. Barbuzza y M. del Fresno, "Tomographic system based on plasma focus x-rays", Nukleonika, Vol. 46, Supplement 1, pp: 5-6, 2001.